



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/729,772	12/08/2003	Erik de Groot	120 05001US	3423
128	7590	01/14/2008		
HONEYWELL INTERNATIONAL INC.			EXAMINER	
101 COLUMBIA ROAD			DAO, THUY CHAN	
P O BOX 2245				
MORRISTOWN, NJ 07962-2245			ART UNIT	PAPER NUMBER
			2192	
			MAIL DATE	DELIVERY MODE
			01/14/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

AK

Office Action Summary	Application No.	Applicant(s)
	10/729,772	DE GROOT ET AL.
	Examiner	Art Unit
	Thuy Dao	2192

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE .03 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 02 November 2007.
- 2a) This action is **FINAL**. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-4 and 6-25 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) _____ is/are allowed.
- 6) Claim(s) 1-4 and 6-25 is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on 08 December 2003 is/are: a) accepted or b) objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) Notice of References Cited (PTO-892)
- 2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____
- 4) Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____
- 5) Notice of Informal Patent Application
- 6) Other: _____

DETAILED ACTION

1. This action is responsive to the amendment filed on November 2, 2007.
2. Claims 1-4 and 6-25 have been examined.

Response to Amendments

3. Per Applicants' request, claims 1-4, 6-8, 10, 12, 14-20, and 22-25 have been amended.

Response to Arguments

4. Applicants' arguments have been fully considered. However, they are not persuasive.

a) Claims 1-4 and 6-21 rejected under 35 USC 35 103(a) as unpatentable over Dardinski in view of Spring (Remarks, pp. 10-12):

i) The Applicants stated, "...Independent claims 1, 12, and 18 have been amended to change all occurrences of 'level' to 'control level' to distinguish from Dardinski's version levels ..." (Remarks, pp. 10-11).

The examiner respectfully disagrees. In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., "control level" is different with --version levels-- or --version control levels--, emphasis added) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

The plain language of the claims merely recites, "...wherein said plurality of control levels comprise a first control level and a second control level that contains one or more features not contained in said first control level ..." (claim 1, lines 8-10, emphasis added).

Per the plain language of the claims, Spring further discloses:

enabling in said source control system a control level of source control from a selection of a plurality of control levels of source control (e.g., a base control

Art Unit: 2192

level which version numbers are assigned manually, col.9: 16-37; an advanced control level which version numbers are changed automatically, col.9: 38-59),

wherein said plurality of control levels comprise a first control level and a second control level that contains one or more features not contained in said first control level (e.g., said two control levels have two different attributes: manually and automatically changed version numbers, col.9: 16-59); and

automatically or manually setting a version number of an object of said source control system, depending on said enabled control level (e.g., col.3: 47 – col.4: 49).

ii) No motivation to combine Dardinski and Spring (Remarks, page 11, second paragraph):

In response to applicant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992).

In this case, Dardinski discloses different Revision levels (col.59: 49-54), Version History (FIG. 52), section 2.1.1.4 Control Levels (col.70-71) and Spring further discloses *automatically or manually setting a version number of a first object of said objects depending on said enabled control level*.

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Spring into that of Dardinski. One would have been motivated to do so to correctly identify different versions of a software module as suggested by Spring (e.g., col.3: 41 – col.4: 49) as well as apply base and/or advanced control levels of source control (e.g., col.9: 16-59).

iii) Hindsight reconstruction of the art (Remarks, pp. 11-12):

Art Unit: 2192

In response to applicant's argument that the examiner's conclusion of obviousness is based upon improper hindsight reasoning, it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the applicant's disclosure, such a reconstruction is proper. See *In re McLaughlin*, 443 F.2d 1392, 170 USPQ 209 (CCPA 1971).

In the instant case, the prior art Spring itself suggests the modification and provides the reason and motivation for making such modification as explicitly disclosed at least in col.3: 41 – col.4: 49 and col.9: 16-59.

b) Claim 22-25 rejected under 35 USC 103(a) as unpatentable over Dardinski in view of Crance and further in view of Spring (Remarks, page 12):

The above discussion of paragraph (a) above over the combination of Dardinski and Spring is fully applicable to amended independent claims 22, 24 and 25 and dependent claim 23.

c) Claims 1, 12, and 18 rejected under 35 USC 103(a) as unpatentable over Dardinski in view of Bowman-Amuah (Remarks, pp. 12-13):

The examiner respectfully disagrees with Applicants' assertions. Bowman-Amuah explicitly teaches:

"Version Control should be set up from the beginning. By delaying version control (i.e., an automated control level has been delayed), manual Version Control must be used (i.e., a manual control level has been used). This result can be an increased cost in disk space in the development environment (because of the number of versions of each module that must be kept) and can lead to some human versioning errors" (col.63: 54-60, emphasis and notation added).

Bowman-Amuah further discloses:

enabling in said source control system a control level of source control from a selection of a plurality of control levels of source control (e.g., a base control level which version numbers are assigned manually and an advanced control level which version numbers are changed automatically, col. 63: 55-60),

wherein said plurality of control levels comprise a first control level and a second control level that contains one or more features not contained in said first control level (e.g., said two control levels have two different attributes: manually and automatically changed version numbers, col. 63: 22-60; col.62: 60 – col.63: 3); and

automatically or manually setting a version number of an object of said source control system, depending on said enabled control level (e.g., col. 63: 22 – col.64: 10).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Bowman-Amuah into that of Dardinski. One would have been motivated to do so to create, maintain, or retrieve multiple versions of source code in a component-based development environment as suggested by Bowman-Amuah (e.g., col.62: 60 – col.63: 31).

d) Claims 22-25 rejected under 35 USC 103(a) as unpatentable over Dardinski in view of Cronce and further in view of Bowman-Amuah (Remarks, page 14):

The above discussion of paragraph (c) above over the combination of Dardinski and Bowman-Amuah is fully applicable to amended independent claims 22, 24 and 25 and dependent claim 23.

Accordingly, the examiner respectfully maintains the 35 USC §103(a) rejection over claims 1-4 and 6-25.

Claim Rejections – 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. Claims 1-4 and 6-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dardinski (art of record, US 6,754,885) in view of Spring (art of record, US Patent No. 6,971,093).

Claim 1:

Dardinski discloses a *method of source control, comprising:*

operating a source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control, wherein said source control system controls versioning of objects (e.g., FIG. 1, Process Control System (source control system) resides on Workstation 11, col.8: 35-43; Controllers 10A-B, col.8: 44-59; FIG. 2, Controllers 10A-B communicates with Devices 12A, Sensors 24 and 26 to provide process control on Valve 18, Tanks 20 and 22, col.9: 20-29);

enabling in said source control system a control level of source control from a selection of a plurality of control levels (e.g., at least two control levels in Revision Levels, col.59: 49-54; Version History, col.59: 56 – col.60: 31); and

automatically or manually setting a version number of a first object of said objects (e.g., an object as Control Algorithm object, col.79: 21-25, col.77: 51-60; FIGs. 46-47, setting version number, col.53: 54 – col.54: 38),

wherein said first object is a control strategy loadable to said controller to provide said process control (e.g., col.2: 25-42; col.9: 33-43).

Dardinski discloses different Revision levels (col.59: 49-54), Version History (FIG. 52), section 2.1.1.4 Control Levels (col.70-71) but does not explicitly disclose [automatically or manually setting a version number of a first object depending on said enabled control level].

However, in an analogous art of version control, Spring further discloses:

enabling in said source control system a control level of source control from a selection of a plurality of control levels of source control (e.g., a base control level which version numbers are assigned manually, col.9: 16-37; an advanced control level which version numbers are changed automatically, col.9: 38-59),

wherein said plurality of control levels comprise a first control level and a second control level that contains one or more features not contained in said first control level (e.g., said two control levels have two different attributes: manually and automatically changed version numbers, col.9: 16-59); and

automatically or manually setting a version number of a first object, depending on said enabled control level (e.g., col.3: 47 – col.4: 49).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Spring into that of Dardinski. One would have been motivated to do so to correctly identify different versions of a software module as suggested by Spring (e.g., col.3: 41 – col.4: 49) as well as apply base and/or advanced control levels of source control (e.g., col.9: 16-59).

Claim 2:

The rejection of claim 1 is incorporated. Dardinski also discloses *providing a capability to switch said enabled control level of source control to another control level of said control levels of source control* (e.g., col.59: 49-54, col.59: 56 – col.60: 31).

Claim 3:

The rejection of claim 1 is incorporated. Dardinski also discloses *automatically setting said version number is based on a degree of change to said first object* (e.g., FIGs. 46-47, col.53: 54 – col.54: 38).

Art Unit: 2192

Claim 4:

The rejection of claim 1 is incorporated. Dardinski also discloses *storing attributes associated with said first object in a database* (e.g., col.79: 21-25; col.77: 51-60).

Claim 6:

The rejection of claim 1 is incorporated. Dardinski also discloses *said plurality of control levels of source control comprises control level none, control level basic, and control level full* (e.g., col.70: 46 – col.71: 24).

Claim 7:

The rejection of claim 1 is incorporated. Dardinski also discloses *for said control level none, said method further comprises: receiving user-entered text for said version number; setting a created-by name set upon receiving a first save changes request; setting a modified-by name upon receiving a save changes request; setting a date-created date upon receiving said first save changes request; and setting a version date upon receiving said save changes request* (e.g., FIG. 48, Revision Editor, col.54: 39-64; FIG. 49: Revision Dialog Box, col.54: 65 – col.55: 14).

Claim 8:

The rejection of claim 1 is incorporated. Dardinski also discloses *for said control level basic, said method further comprises: automatically incrementing said version number upon receiving a save changes request, including a first save changes request; setting a created-by name upon receiving said first save changes request; setting a modified-by name upon receiving said save changes request, including a first save changes request; setting a date-created date upon receiving said first save changes request; setting a version date upon receiving said save changes request, including a first save changes request; and displaying said version number* (e.g., FIG. 50: col.55; 23-63).

Art Unit: 2192

Claim 9:

The rejection of claim 1 is incorporated. Dardinski also discloses *said version number is incremented differently for minor changes than for major changes* (e.g., col.55: 65 – col.56: 48).

Claim 10:

The rejection of claim 1 is incorporated. Dardinski also discloses *for said control level full, said method further comprises: supporting a qualification life cycle model; providing a version control system toolbar and menu; automatically incrementing said version number upon check-in, including a first check-in wherein said version number is generated; displaying said version number; setting a created-by name upon said first check-in; setting a modified-by name upon said check-in, including said first check-in; in; setting a date-created date upon said check-in, including said first check- setting a version date upon said check-in; receiving a check-in comment; and providing a version history and audit trail* (e.g., FIGs. 52-53: col.59: 57 – col.60: 31).

Claim 11:

The rejection of claim 10 is incorporated. Dardinski also discloses *said version number is incremented differently for minor changes than for major changes, according to user preferences* (e.g., col.55: 65 – col.56: 48).

Claim 12:

Dardinski discloses a process control system, comprising:

a network coupling said computer to a controller that communicates with one or more devices to provide process control (e.g., FIG. 1, Network 14, Workstation 11, Controllers 10A-B, Device 12, col.8: 23-59; FIG. 2, process control in blocks 29-32 to control Valve 18, Tanks 20 and 22, col.9: 20-29);

said computer comprises a source control system with a selectable control level of source control, wherein said source control system controls versioning of at least one control strategy for said process control (e.g., selectable control level of

source control as Control Levels, col.70:32-39; col.70: 46 – col.71: 25; a control strategy as Control Algorithm object, col.2: 25-42, col.9: 33-43); and

wherein said at least one control strategy in said source control system is loadable from said computer to said controller to provide said process control according to said control strategy (e.g., Control Algorithm object (control strategy) is downloaded to field controllers/devices to provide control process on Valve 18, Tanks 20 and 22, col. 2: 25-42; col.9: 33-43, 20-29).

Dardinski discloses different Revision levels (col.59: 49-54), Version History (FIG. 52), section 2.1.1.4 Control Levels (col.70-71) but does not explicitly disclose a *plurality of control levels of source control, wherein said plurality of control levels comprise a first control level and a second control level that contains one or more features not contained in said first control level.*

However, in an analogous art of version control, Spring further discloses:

enabling in said source control system a control level of source control from a selection of a plurality of control levels of source control (e.g., a base control level which version numbers are assigned manually, col.9: 16-37; an advanced control level which version numbers are changed automatically, col.9: 38-59),

wherein said plurality of control levels comprise a first control level and a second control level that contains one or more features not contained in said first control level (e.g., said two control levels have two different attributes: manually and automatically changed version numbers, col.9: 16-59).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Spring into that of Dardinski. One would have been motivated to do so to correctly identify different versions of a software module as suggested by Spring (e.g., col.3: 41 – col.4: 49) as well as apply base and/or advanced control levels of source control (e.g., col.9: 16-59).

Claim 13:

The rejection of claim 12 is incorporated. Dardinski also discloses a *database to store source control information associated with said at least one control strategy, including a version number* (e.g., col.79: 21-25; col.77: 51-60).

Claim 14:

The rejection of claim 13 is incorporated. Dardinski also discloses *said selectable control level of source control is no source control and further wherein a version number is entered manually when said at least one control strategy is saved* (e.g., col.54: 39-64).

Claim 15:

The rejection of claim 13 is incorporated. Dardinski also discloses *said selectable control level of source control is basic source control and further wherein a version number is automatically incremented when said at least one control strategy is saved* (e.g., col.54: 65 – col.55: 14).

Claim 16:

The rejection of claim 13 is incorporated. Dardinski also discloses *said selectable control level of source control is full source control and further wherein a version number is automatically incremented when said at least one control strategy is checked-in* (e.g., col.55: 23-63).

Claim 17:

The rejection of claim 12 is incorporated. Dardinski also discloses *said selectable control level of source control is selected from the group consisting of: a preference, a license, an installation configuration, and a user interface* (e.g., col.70: 32-39; col.70: 46 – col.71: 25).

Claim 18:

As set forth in claims 1 and 12, Dardinski discloses a method for providing a source control system for a process control system, comprising: operating said source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control, wherein said source control system controls versioning of objects; receiving in said source control system a selection from at least two control levels of source control for an object of said source control system; providing a user-enterable version number or an automatically incremented version number when said first object is stored, wherein said object is a control strategy loadable to said controller to provide said process control.

Dardinski does not explicitly disclose providing a user-enterable version number when said first object is stored, if said selection is a first control level; and providing an automatically incremented version number when an-said object is stored, if said selection is a second control level.

However, in an analogous art of version control, Spring further discloses:

a plurality of control levels of source control (e.g., col.9: 16-59);

providing a user-enterable version number when said object is stored, if said selection is a first control level and providing an automatically incremented version number when said object is stored, if said selection is a second control level (e.g., col.9: 16-46, manually setting a version number when module is newly created (first control level) and automatically setting a version number when said module is modified/updated (second control level)).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Spring into that of Dardinski. One would have been motivated to do so to correctly identify different versions of a software module as suggested by Spring (e.g., col.3: 41 – col.4: 49) as well as apply base and/or advanced control levels of source control (e.g., col.9: 16-59).

Claim 19:

The rejection of claim 18 is incorporated. Dardinski also discloses *providing an automatically incremented version number when said first object is checked-in, if said selection is a third control level of said plurality of control levels of source control* (e.g., col.55: 23-63).

Claim 20:

The rejection of claim 18 is incorporated. Dardinski also discloses *changing said selection to another of said plurality of control levels of source control* (e.g., col.54: 39-64; col.54: 65 – col.55: 14).

Claim 21:

The rejection of claim 18 is incorporated. Dardinski also discloses *updating attributes of said object based on said selection* (e.g., col.53; 54 – col.54: 38).

7. Claims 22-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dardinski in view of Cronce (art of record, US Patent Publication No. 2003/0156719 A1) and further in view of Spring.

Claim 22:

Dardinski discloses a *computer readable medium having executable instructions stored thereon to perform a method of providing configurable control levels of support for a source control system, said method comprising:*

operating said source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control (e.g., FIG. 1, col.8: 23-59);

at least one control strategy of said source control system; wherein said at least one control strategy is loadable from said computer to said controller to provide said process control according to said at least one control strategy (e.g., col.9: 20-29; col.2: 25: 42; col.9: 33-43).

Dardinski does not explicitly disclose *receiving a request for a control level of support, determining whether a full control level of support is licensed, determining whether an option for a basic control level of support is selected, setting said control level of support to full, if said full control level of support is licensed, and setting said control level of support to basic is said option is selected.*

However, in an analogous art of providing licensed software, Crone further discloses *receiving a request for a control level of support, determining whether a full control level of support is licensed, determining whether an option for a basic control level of support is selected, setting said control level of support to full, if said full control level of support is licensed, and setting said control level of support to basic is said option is selected* (e.g., [0002], [0037], [0058], [0060]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Crone into that of Dardinski. One would have been motivated to do so to provide licensed software and control said licensed software usage based on full, partial, basic, or trial levels as suggested by Crone (e.g., [0002], [0010-0012]).

Neither Dardinski nor Crone explicitly discloses a *plurality of control levels of source control including a full control level and a basic control level.*

However, in an analogous art of version control, Spring further discloses a *plurality of control levels of source control* (e.g., col.9: 16-59) *including a full control level* (e.g., col.9: 38-59) *and a basic control level* (e.g., col.9: 16-37).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Spring into that of Dardinski. One would have been motivated to do so to correctly identify different versions of a software module as suggested by Spring (e.g., col.3: 41 – col.4: 49) as well as apply base and/or advanced control levels of source control (e.g., col.9: 16-59).

Claim 23:

The rejection of claim 22 is incorporated. Cponce further discloses a *default for said control level of support is none* (e.g., [0037], [0058]).

Claim 24:

Dardinski discloses a *computer readable medium having executable instructions stored thereon to perform a method of changing configurable control levels of support for a source control system, said method comprising:*

operating said source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control, wherein said source control system controls versioning of objects (e.g., FIG. 1, col.8: 35-59; FIG. 2, col.9: 20-29);

said first object of said objects, which is loadable from said computer to said controller to provide said process control according to said object (e.g., col.2: 25-42; col.9: 33-43).

Cponce further discloses *receiving a request from a user to change a control level, determining whether a full control level is licensed, determining whether said request is to change from none to basic, determining whether said request is to change from basic to none, performing said request when said request is to change from none to basic or from basic to none, and storing a new control level* (e.g., [0002], [0037], [0058], [0060], full, partial, basic, trial levels; user stops using software after trial period (none) or trial period expires).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Cponce into that of Dardinski. One would have been motivated to do so to provide licensed software and control said licensed software usage based on full, partial, basic, or trial levels as suggested by Cponce (e.g., [0002], [0010-0012]).

Neither Dardinski nor Cponce explicitly discloses a *plurality of control levels of source control including a full control level, and a basic/none control level.*

However, in an analogous art of version control, Spring further discloses a *plurality of control levels of source control* (e.g., col.9: 16-59) *including a full control level* (e.g., col.9: 38-59) *and a basic/none control level* (e.g., col.9: 16-37).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Spring into that of Dardinski. One would have been motivated to do so to correctly identify different versions of a software module as suggested by Spring (e.g., col.3: 48 – col.4: 49) as well as apply base/none and/or advanced control levels of source control (e.g., col.9: 16-59).

Claim 25:

Dardinski discloses a *computer readable medium having executable instructions stored thereon to perform a method of updating version attributes based on a control level of source control, said method comprising:*

operating said source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control, wherein said source control system controls versioning of objects (e.g. FIG. 1, col.8: 35-59);

said first object of said objects (e.g., col.2: 25-42);

determining whether said first object is new (e.g., col.53: 54 – col.54: 38);

setting a version number to a first version number, when said first object is new (e.g., FIG. 52, col.59: 57 – col.60: 5);

updating version attributes of said first object (e.g., col.53: 54 – col.54: 38); and

incrementing said version number, when said first object is not new, wherein said first object is loadable from said computer to said controller to provide said process control according to said object (e.g., FIG. 45, col.52: 7-14; col.59: 57 – col.60: 5).

Cronce further discloses *determining whether an object is licensed, determining whether a basic control level is selected, receiving a save changes request for said object, updating version attributes of said object according to whether said full control*

level is licensed and whether said basic control level is selected (e.g., [0002], [0037], [0058], [0060]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Crone into that of Dardinski. One would have been motivated to do so to provide licensed software and control said licensed software usage based on full, partial, basic, or trial levels as suggested by Crone (e.g., [0002], [0010-0012]).

Neither Dardinski nor Crone explicitly discloses a *plurality of control levels of source control including a full control level, and a basic control level.*

However, in an analogous art of version control, Spring further discloses a *plurality of control levels of source control (e.g., col.9: 16-59) including a full control level (e.g., col.9: 38-59) and a basic control level (e.g., col.9: 16-37).*

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Spring into that of Dardinski. One would have been motivated to do so to correctly identify different versions of a software module as suggested by Spring (e.g., col.3: 41 – col.4: 49) as well as apply base and/or advanced control levels of source control (e.g., col.9: 16-59).

8. Claims 1, 12, and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dardinski in view of Bowman-Amuah (art of record, US Patent No. 6,662,357).

Claim 1:

Dardinski discloses a *method of source control, comprising:*

operating a source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control, wherein said source control system controls versioning of objects (e.g., FIG. 1, Process Control System (source control system) resides on Workstation 11, col.8: 35-43; Controllers 10A-B, col.8: 44-59; FIG. 2, Controllers 10A-B communicates with Devices 12A, Sensors 24 and 26 to provide process control on Valve 18, Tanks 20 and 22, col.9: 20-29);

enabling in said source control system a control level of source control from a selection of a plurality of control levels (e.g., at least two control levels in Revision Levels, col.59: 49-54; Version History, col.59: 56 – col.60: 31); and

automatically or manually setting a version number of a first object of said objects (e.g., an object as Control Algorithm object, col.79: 21-25, col.77: 51-60; FIGs. 46-47, setting version number, col.53: 54 – col.54: 38),

wherein said first object is a control strategy loadable to said controller to provide said process control (e.g., col.2: 25-42; col.9: 33-43).

Dardinski discloses different Revision levels (col.59: 49-54), Version History (FIG. 52), section 2.1.1.4 Control Levels (col.70-71) but does not explicitly disclose [*automatically or manually setting a version number of an object of said source control system] depending on said enabled control level.*

However, in an analogous art of version control, Bowman-Amuah further discloses:

enabling in said source control system a control level of source control from a selection of a plurality of control levels of source control (e.g., a base control level which version numbers are assigned manually and an advanced control level which version numbers are changed automatically, col. 63: 55-60),

wherein said plurality of control levels comprise a first control level and a second control level that contains one or more features not contained in said first control level (e.g., said two control levels have two different attributes: manually and automatically changed version numbers, col. 63: 22-60; col.62: 60 – col.63: 3); and

automatically or manually setting a version number of a first object of said objects, depending on said enabled control level (e.g., col. 63: 22 – col.64: 10).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Bowman-Amuah into that of Dardinski. One would have been motivated to do so to create, maintain, or retrieve multiple versions of source code in a component-based development environment as suggested by Bowman-Amuah (e.g., col.62: 60 – col.63: 31).

Claim 12:

Dardinski discloses a process control system, comprising:

a network coupling said computer to a controller that communicates with one or more devices to provide process control (e.g., FIG. 1, Network 14, Workstation 11, Controllers 10A-B, Device 12, col.8: 23-59; FIG. 2, process control in blocks 29-32 to control Valve 18, Tanks 20 and 22, col.9: 20-29);

said computer comprises a source control system with a selectable control level of source control for at least one control strategy, wherein said source control system controls versioning of at least one control strategy for said process control (e.g., selectable control level of source control as Control Levels, col.70:32-39; col.70: 46 – col.71: 25; a control strategy as Control Algorithm object, col.2: 25-42, col.9: 33-43); and

wherein said at least one control strategy in said source control system is loadable from said computer to said controller to provide said process control according to said control strategy (e.g., Control Algorithm object (control strategy) is downloaded to field controllers/devices to provide control process on Valve 18, Tanks 20 and 22, col. 2: 25-42; col.9: 33-43, 20-29).

Dardinski discloses different Revision levels (col.59: 49-54), Version History (FIG. 52), section 2.1.1.4 Control Levels (col.70-71) but does not explicitly disclose a plurality of control levels of source control, wherein said plurality of control levels comprise a first control level and a second control level that contains one or more features not contained in said first control level.

However, in an analogous art of version control, Bowman-Amuah further discloses:

enabling in said source control system a control level of source control from a selection of a plurality of control levels of source control (e.g., a base control level which version numbers are assigned manually and an advanced control level which version numbers are changed automatically, col.63: 55-60),

wherein said plurality of control levels comprise a first control level and a second control level that contains one or more features not contained in said first control level (e.g., said two control levels have two different attributes: manually and automatically changed version numbers, col.63: 22-60; col.62: 60 – col.63: 3).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Bowman-Amuah into that of Dardinski. One would have been motivated to do so to as set forth above.

Claim 18:

As set forth in claims 1 and 12, Dardinski discloses a *method for providing a source control system for a process control system, comprising: operating said source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control, wherein said source control system controls versioning of at least one control strategy for said process control; receiving in said source control system a selection from at least two control levels of source control for an object of said source control system; providing a user-enterable version number or an automatically incremented version number when an-said object is stored, wherein said object is a control strategy loadable to said controller to provide said process control.*

Dardinski does not explicitly disclose *providing a user-enterable version number when said first object is stored, if said selection is a first control level; and providing an automatically incremented version number when said first object is stored, if said selection is a second control level.*

However, in an analogous art of version control, Bowman-Amuah further discloses:

a plurality of control levels of source control (e.g., col.63: 55-60);
providing a user-enterable version number when said first object is stored, if said selection is a first control level and providing an automatically incremented version number when said object is stored, if said selection is a second control level (e.g., col.63: 22-60; col.62: 60 – col.63: 3).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Bowman-Amuah into that of Dardinski. One would have been motivated to do so as set forth in claim 1 above.

9. Claims 22 and 24-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dardinski in view of Crone and further in view of Bowman-Amuah.

Claim 22:

Dardinski discloses a computer readable medium having executable instructions stored thereon to perform a method of providing configurable control levels of support for a source control system, said method comprising:

operating said source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control (e.g., FIG. 1, col.8: 23-59);

at least one control strategy of said source control system; wherein said at least one control strategy is loadable from said computer to said controller to provide said process control according to said at least one control strategy (e.g., col.9: 20-29; col.2: 25: 42; col.9: 33-43).

Dardinski does not explicitly disclose receiving a request for a control level of support, determining whether a full control level of support is licensed, determining whether an option for a basic control level of support is selected, setting said control level of support to full, if said full control level of support is licensed, and setting said control level of support to basic if said option is selected.

However, in an analogous art of providing licensed software, Crone further discloses receiving a request for a control level of support, determining whether a full control level of support is licensed, determining whether an option for a basic control level of support is selected, setting said control level of support to full, if said full control level of support is licensed, and setting said control level of support to basic if said option is selected (e.g., [0002], [0037], [0058], [0060]).

Art Unit: 2192

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Crone into that of Dardinski. One would have been motivated to do so to provide licensed software and control said licensed software usage based on full, partial, basic, or trial control levels as suggested by Crone (e.g., [0002], [0010-0012]).

Neither Dardinski nor Crone explicitly discloses a *plurality of control levels of source control including a full control level and a basic control level.*

However, in an analogous art of version control, Bowman-Amuah further discloses a *plurality of control levels of source control including a full control level and a basic control level* (e.g., col.63: 22-60; col.62: 60 – col.63: 3).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Bowman-Amuah into that of Dardinski. One would have been motivated to do so to as set forth in claim 1 above.

Claim 24:

Dardinski discloses a *computer readable medium having executable instructions stored thereon to perform a method of changing configurable control levels of support for a source control system, said method comprising:*

operating said source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control, wherein said source control system controls versioning of at least one control strategy for said process control (e.g., FIG. 1, col.8: 35-59; FIG. 2, col.9: 20-29);

an object of source control, which is loadable from said computer to said controller to provide said process control according to said object (e.g., col.2: 25-42; col.9: 33-43).

Crone further discloses *receiving a request from a user to change a control level, determining whether a full control level is licensed, determining whether said request is to change from none to basic, determining whether said request is to change from basic to none, performing said request when said request is to change from none*

Art Unit: 2192

to basic or from basic to none, and storing a new control level (e.g., [0002], [0037], [0058], [0060], full, partial, basic, trial levels; user stops using software after trial period (none) or trial period expires).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Cponce into that of Dardinski. One would have been motivated to do so to provide licensed software and control said licensed software usage based on full, partial, basic, or trial levels as suggested by Cponce (e.g., [0002], [0010-0012]).

Neither Dardinski nor Cponce explicitly discloses a *plurality of control levels of source control including a full control level, and a basic/none control level.*

However, in an analogous art of version control, Bowman-Amuah further discloses a *plurality of control levels of source control including a full control level and a basic/none control level* (e.g., col.63: 22-60; col.62: 60 – col.63: 3).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Bowman-Amuah into that of Dardinski. One would have been motivated to do so to as set forth above.

Claim 25:

Dardinski discloses a *computer readable medium having executable instructions stored thereon to perform a method of updating version attributes based on a control level of source control, said method comprising:*

operating said source control system on a computer that is coupled via a network to a controller that communicates with one or more devices to provide process control (e.g. FIG. 1, col.8: 35-59);

a first object of said objects (e.g., col.2: 25-42);

determining whether said first object is new (e.g., col.53: 54 – col.54: 38);

setting a version number to a first version number, when said first object is new (e.g., FIG. 52, col.59: 57 – col.60: 5);

updating version attributes of said first object (e.g., col.53: 54 – col.54: 38); and

incrementing said version number, when said object is not new, wherein said object is loadable from said computer to said controller to provide said process control according to said object (e.g., FIG. 45, col.52: 7-14; col.59: 57 – col.60: 5).

Cronce further discloses *determining whether an object is licensed, determining whether a basic control level is selected, receiving a save changes request for said object, updating version attributes of said object according to whether said full control level is licensed and whether said basic control level is selected* (e.g., [0002], [0037], [0058], [0060]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Cronce into that of Dardinski. One would have been motivated to do so to provide licensed software and control said licensed software usage based on full, partial, basic, or trial levels as suggested by Cronce (e.g., [0002], [0010-0012]).

Neither Dardinski nor Cronce explicitly discloses *a plurality of control levels of source control including a full control level, and a basic control level.*

However, in an analogous art of version control, Bowman-Amuah further discloses *a plurality of control levels of source control including a full control level and a basic control level* (e.g., col.63: 22-60; col.62: 60 – col.63: 3).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine the teaching of Bowman-Amuah into that of Dardinski. One would have been motivated to do so to as set forth in claim 1 above.

Conclusion

10. Applicants' amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

11. Any inquiry concerning this communication should be directed to examiner Thuy Dao (Twee), whose telephone/fax numbers are (571) 272 8570 and (571) 273 8570, respectively. The examiner can normally be reached on every Tuesday, Thursday, and Friday from 6:00AM to 6:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam, can be reached at (571) 272 3695.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273 8300.

Any inquiry of a general nature of relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is (571) 272 2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

T. Dao


TUAN DAM
SUPERVISORY PATENT EXAMINER